Agile Methodologies

Mrunmayee Gondhalekar[1], Yogeshchandra Puranik[2]

[1]PG Student, [2]Assistant Professor,
[1,2]Affiliated to Department of (MCA), P.E.S. Modern College of Engineering, Pune, Maharashtra, India

ABSTRACT

Agile software development methodologies have gained traction in the recent past since it was introduced in the mid-'90s. They have also been highly adopted in the software development group. In the past 15 years, there have been enormous researches which have been done on agile methodologies and surveys done on the methods of agile methods. In this paper, a survey of various studies on agile methodologies will be conducted using a 'compare and review' review method. There is also a mention of the other agile methods which have not been widely researched.

KEYWORDS: *agile software development, agile methodologies, XP, Scrum, SDLC*

1. INTRODUCTION:

Agile methodology is a software design and development approach that is iterative, lean, and lightweight that was borne in the 1990s in tandem with the call to have rapid software development to work well with the changing business environment. The business environment was changing to the need to have now a fast way of developing software. It is comparable to climbing a ladder that has short steps, which have the exact distance between the steps. In the same way, agile methodology divides the task into smaller iterations that have the same length. The transition between the iterations is smoother and faster than when we had to develop the entire software system.

Agile processes try to find an equilibrium so that the software development is done fast and in a high-quality approach.

In a dynamic business environment, requirements change fast, and at the same time, there is a need to develop high software quality products. Different from other methods, agile methodology has customer feedback as the control mechanism. This approach ensures high customer satisfaction.

The agile methodology encompasses various methodologies like Extreme Programming (XP), Adaptive Software Development (ASD), Scrum, Dynamic System Development Methodologies (DSDM), Feature Driven Programming, Crystal methods, and Agile Unified Process (AUP). There are other variants of agile methodologies which have been developed.

The agile methodology is based on the iterative improvement where each iteration in the development represents a small scale and self-contained software development life cycle (SDLC). Different from the spiral method in which it espouses simplicity in all the entire software development processes.

This paper focuses on a survey of agile methodologies and the development of new methodology based on compare and review approach.

The agile methods have been divided into different categories so that the study can exhaustively compare the agile methods presented. Four categories of agile methods have been highlighted in the paper: agile requirements engineering, Hybrid agile methods, Agile methods, and extra.

2. Methodology

The study followed a two-step procedure, where the first one was where the researcher aimed to compare agile methodologies. At the same time, the second step was to look at the review of the latest agile methodologies, which have had a little review in literature.

2.1. Research requirements

In this section, the research requirements needed to have a research in this study were evaluated.

One of the requirements was to have a basic understanding of the agile software development methodologies. Develop a comprehensive grasp of such methodologies as XP, Scrum, and FDD. Another requirement was to have a clear comparison of the changes that have been noted in the new agile methodologies so that they are differentiated from the SDLC approaches that have been suggested before.

2.2. Method of selecting literature

The search and selection criteria used include sourcing from approved digital databases like ACM digital library, IEEE digital library, Science direct and university portal. The research material is either a conference paper or a journal, the period between 2017 and 2021. The research articles must have been published in the English language. The paper has to have one of the following keywords: Agile software methods, Extreme Programming, Scrum, and survey.

2.3. Survey of agile methods

This stage entails a survey of all research studies that have been conducted on different agile methodologies. The year of publications for these surveys must have been between 2017 to 2021.

A rigorous study will be conducted so that the correct information and proper search criteria will be used at the end.

2.4. Literature review of agile methodologies

In this section, the authors had a review of 10 papers that have not been assessed and mention yet in previous literature reviews that have been done.

3. Results

From the survey of the literature, four categories of agile methodologies have been identified.

3.1. Agile requirements engineering

One of the beauties of agile methodology is that changes can be incorporated at any stage of software development in the SDLC, making it suitable for dynamic business environments where changes are frequently experienced. There have been different ways in which requirements have been managed and enhanced in agile engineering methodologies, as advocated by (Baruah, 2015). There are different ways in which requirements have been managed. Eight agile methodologies were compared in terms of how the requirements have been managed.

Another article by (Inayat et al.,2015)

Presents a comprehensive review about ways in which traditional requirements engineering issues can be solved with the use of agile requirements technologies. The studies were done between 2002 and 2013. 21 out of 531 papers were studied in this process. The study's main results can be stated as 57% conferences, 19% as journals, 5% as magazines.

Four methods can be identified in the study. The four methods include agile requirements engineering, new methods in agile methodologies, traditional requirements engineering compared to agile requirements engineering, and agile requirements engineering in general. There is also an empirical study on agile requirements engineering practices (Ramesh et al.,2010). The data was collected from engineering companies located in the United States so that they would be able to compare traditional requirements engineering and agile requirements engineering. The solicitation of the data was achieved through systematic literature reviews. There has also been a framework that has been developed that would help to assess the risks that are associated with using agile requirements engineering.

Six methods of requirements engineering were identified from the studies, which include face-to-face communication instead of written specifications. There are also iterative requirements engineering. The third is the use of Extreme

practices while looking for requirements. There is also prototyping. The lasts are the use of review meetings with acceptance tests.

Problems identified in agile requirements engineering –

Some problems were identified, which include: challenges in cost estimations, unsuitable architecture, oversight of non-functional requirements, customer participation which complicates the process, prioritizing a single dimension, insufficient method of verifying requirements, and not enough documentation.

The risks identified in using agile requirements engineering include lack of stable requirements, challenges with customer consent and capacity, not enough iterations between the customer and developer, absent significant requirements, modelling only functional requirements, oversight of requirements reviewing, making use of designs to present the requirements, focusing so much on perfecting requirements before coding stages started, and significant flaws in schedules.

3.2. Agile methods

A survey of agile methods was done between 2011 and 2010 (STAVRU, 2014). The following survey suggested that the quality of the papers done in this period could still be improved. The suggestions in most of the papers are that there is a need to compare agile methods with other methods and assess the rate of use of agile methods in the levels of an organization.

Also, there is a need to study the adaptation of agile methods by academia so that academia and industry do not have too wide a gap.

There is also a need to provide detailed reports in future research to increase confidence. An agile method tailoring is also an aspect of agile methodology where an agile methodology is selected to be in an organization. Most of the studies focus on the types of agile methods selected and the research validation that has been adopted (Campanelli and Parreiras,2015). Another study showed the agile research methodologies which have been done in the past. They include the principles of agile methodology and the degree of agility, which include adopting agile regarding the needed changes. The second aspect of primary concern here is the research o agile software development where agile software development processes are espoused (Dingsoyr et al.,2012). The article also showed the seminal contribution and the relationships between the sources and the seminal information.

Another study focuses on Distributed pair programming (DPP) from the both academic and industrial perspective. Pair programming (PP) means two programmers are working on the same project but in different locations.

3.3. Hybrid Agile methods

A review study was done on agile methodologies, which an intricate capability maturity model (CMM). There was an overview of the benefits that this has on the development process and the organization in general. The results show that using agile methods is helpful to reach level 2 and, in some instances, even level 3 of CMM (Silva et al.,2015).

Web development companies have for a long time have used the CMM – DEV model. They have followed one agile method for a long time. The result shows that in the recent five years,

more and more companies are shifting to agile methods to get certified against CMM-DEV.

3.4. Extra Agile

The results show that scrum is the most commonly used agile method. Extreme follows a distant second (Rauf and AlGhafees,2015). Similar research was done at the Microsoft software where employees showed that they preferred scrum. Most of the agile users have a favourable view of the development's agile methodology (Begel and Nagappan, 2007).

4. Discussion

Agile methodology resulted from dissatisfaction with Plan Driven software development because when small systems were developed using a planned approach, there would be unnecessary overhead cost. Software development using traditional approaches dominated the software development process. There is more time sent on how the system should be developed and not on the actual development of the system.

When requirements change, the software system also has to be changed, and the design must be reworked again. The specification and design have to change with the design. It is out of this that software engineers proposed agile methods who were dissatisfied with the heavyweight software development processes.

Businesses now operate in a dynamic environment where new requirements in their business operations are met every day.

Conclusion

Agile is one of the most popular software design and development methodologies researched in the world today. In this study, a survey of different agile methods has been reviewed to get to know how the new methods work.

References

[1] Baruah, N. (2015). Requirement management in the agile software environment. *Procedia Computer Science*, *62*, 81–83.

[2] Begel, A., & Nagappan, N. (2007). Usage and perceptions of agile software development in an industrial context: An exploratory study. *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)*, 255–264.

[3] Campanelli, A. S., & Parreiras, F. S. (2015). Agile methods tailoring–A systematic literature review. *Journal of Systems and Software*, *110*, 85–100.

[4] Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). *A decade of agile methodologies: Towards explaining agile software development*. Elsevier.

[5] Inayat, I., Salim, S. S., Marczak, S., Daneva, M., & Shamshirband, S. (2015). A systematic literature review on agile requirements engineering practices and challenges. *Computers in Human Behavior*, *51*, 915–929.

[6] Ramesh, B., Cao, L., & Baskerville, R. (2010). Agile requirements engineering practices and challenges: an empirical study. *Information Systems Journal*, *20*(5), 449–480.

[7] Rauf, A., & AlGhafees, M. (2015). Gap Analysis between State of Practice and State of Art Practices in Agile Software Development. *2015 Agile Conference*, 102–106.

[8] Santana, C., Queiroz, F., Vasconcelos, A., & Gusmão, C. (2015). Software process improvement in agile software development a systematic literature review. *2015 41st Euromicro Conference on Software Engineering and Advanced Applications*, 325–332.

[9] Silva, F. S., Soares, F. S. F., Peres, A. L., de Azevedo, I. M., Vasconcelos, A. P. L. F., Kamei, F. K., & de Lemos Meira, S. R. (2015). Using CMMI together with agile software development: A systematic review. *Information and Software Technology*, *58*, 20–43.

[10] Stavru, S. (2014). A critical examination of recent industrial surveys on agile method usage. *Journal of Systems and Software*, *94*, 87–97.

[11] Torrecilla-Salinas, C. J., Sedeño, J., Escalona, M. J., & Mejías, M. (2016). Agile, Web Engineering and Capability Maturity Model Integration: A systematic literature review. *Information and Software Technology*, *71*, 92–107.